Notice of Allowability	Application No.	Applicant(s)	
	09/748,862	SAKAMOTO ET AL.	
	Examiner	Art Unit	
	Jennifer N. To	2195	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address--

All claims being allowable, PROSECUTION ON THE MERITS IS (OR REMAINS) CLOSED in this application. If not included herewith (or previously mailed), a Notice of Allowance (PTOL-85) or other appropriate communication will be mailed in due course. **THIS NOTICE OF ALLOWABILITY IS NOT A GRANT OF PATENT RIGHTS.** This application is subject to withdrawal from issue at the initiative of the Office or upon petition by the applicant. See 37 CFR 1.313 and MPEP 1308.

1. ☒ This communication is responsive to *03/31/2006, email amendment 05/09/2006*.

2. ☒ The allowed claim(s) is/are *16-24, now renumbered as 1-9*.

3. ☒ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).
   a) ☒ All b) ☐ Some* c) ☐ None of the:
      1. ☐ Certified copies of the priority documents have been received.
      2. ☐ Certified copies of the priority documents have been received in Application No. _____ .
      3. ☐ Copies of the certified copies of the priority documents have been received in this national stage application from the International Bureau (PCT Rule 17.2(a)).

   * Certified copies not received: _____.

Applicant has THREE MONTHS FROM THE "MAILING DATE" of this communication to file a reply complying with the requirements noted below. Failure to timely comply will result in ABANDONMENT of this application.
THIS THREE-MONTH PERIOD IS NOT EXTENDABLE.

4. ☐ A SUBSTITUTE OATH OR DECLARATION must be submitted. Note the attached EXAMINER'S AMENDMENT or NOTICE OF INFORMAL PATENT APPLICATION (PTO-152) which gives reason(s) why the oath or declaration is deficient.

5. ☐ CORRECTED DRAWINGS (as "replacement sheets") must be submitted.
   (a) ☐ including changes required by the Notice of Draftsperson's Patent Drawing Review (PTO-948) attached
      1) ☐ hereto or 2) ☐ to Paper No./Mail Date _____.
   (b) ☐ including changes required by the attached Examiner's Amendment / Comment or in the Office action of Paper No./Mail Date _____.

Identifying indicia such as the application number (see 37 CFR 1.84(c)) should be written on the drawings in the front (not the back) of each sheet. Replacement sheet(s) should be labeled as such in the header according to 37 CFR 1.121(d).

6. ☐ DEPOSIT OF and/or INFORMATION about the deposit of BIOLOGICAL MATERIAL must be submitted. Note the attached Examiner's comment regarding REQUIREMENT FOR THE DEPOSIT OF BIOLOGICAL MATERIAL.

Attachment(s)

1. ☒ Notice of References Cited (PTO-892)
2. ☐ Notice of Draftperson's Patent Drawing Review (PTO-948)
3. ☒ Information Disclosure Statements (PTO-1449 or PTO/SB/08), Paper No./Mail Date 03/31/2006
4. ☐ Examiner's Comment Regarding Requirement for Deposit of Biological Material
5. ☐ Notice of Informal Patent Application (PTO-152)
6. ☒ Interview Summary (PTO-413), Paper No./Mail Date _____ .
7. ☒ Examiner's Amendment/Comment
8. ☐ Examiner's Statement of Reasons for Allowance
9. ☐ Other _____.

SUPERVISORY PATENT EXAMINER
2100

EXAMINER'S AMENDMENT

1. An examiner's amendment to the record appears below. Should the changes and/or additions be unacceptable to applicant, an amendment may be filed as provided by 37 CFR 1.312. To ensure consideration of such an amendment, it MUST be submitted no later than the payment of the issue fee.

2. Authorization for this examiner's amendment was given in a telephone interview with Ms. Allison Olengski on 05/08/2006.

3. The claims had been amended in accordance to the attachment email received on 05/09/2006.

4. Any inquiry concerning this communication or earlier communications from the examiner should be directed to Jennifer N. To whose telephone number is (571) 272-7212. The examiner can normally be reached on M-T 6AM- 3:30 PM, F 6AM- 2:30 PM.

5. If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Meng-Ai An can be reached on (571) 272-3756. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

6. Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for

published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

Jennifer N To
Examiner
Art Unit 2195

SUPERVISORY PATENT EXAMINER
TECHNOLOGY CENTER 2100

The text of all pending claims, (including withdrawn claims) is set forth below. Cancelled and not entered claims are indicated with claim number and status only. The claims as listed below show added text with underlining and deleted text with ~~strikethrough~~. The status of each claim is indicated with one of (original), (currently amended), (cancelled), (withdrawn), (new), (previously presented), or (not entered). Please AMEND claims 16, 17, 19, 20, 22, and 23 in accordance with the following:

Claims 1-15 (Cancelled).

16. (CURRENTLY AMENDED) A method for controlling a plurality of threads that perform parallel processing, comprising:

counting a number of running threads performing parallel processing using a running thread counter and a number of standby threads that are in a standby state using a standby thread counter for a predetermined time period;

setting a maximum number of running threads in accordance with the number of running threads during the predetermined time period using a maximum running thread counter;

comparing the number of standby threads with the maximum number of running threads;

terminating a number of standby threads exceeding the maximum number of running threads when the number of standby threads is greater than the maximum number of running threads;

incrementing the number of running threads using the running thread counter and decrementing the number of standby threads using the standby thread counter in response to a run request; and

updating the maximum number of running threads set by the maximum running thread counter if the incremented number of running threads is greater than the maximum number of running threads.

17. (CURRENTLY AMENDED) A method for controlling a plurality of threads that perform parallel processing, comprising:

counting a number of running threads performing parallel processing using a running thread counter and a number of standby threads that are in a standby state using a standby thread counter for a predetermined time period;

setting an average number of running threads in accordance with the number of running threads during the predetermined time period using an average running thread counter;

comparing the number of standby threads with the average number of running threads; and

terminating a number of standby threads exceeding the average number of running threads when the number of standby threads is greater than the average number of running threads.

18. (PREVIOUSLY PRESENTED) A method for controlling a plurality of threads that perform parallel processing, comprising:

counting a number of running threads performing parallel processing using a running thread counter and a number of standby threads that are in a standby state using a standby thread counter for a predetermined time period;

setting a product obtained by multiplying the number of running threads during the predetermined time period by a predetermined coeficient using a product running thread counter;

comparing the number of standby threads with the product;

terminating a number of standby threads exceeding the product when the number of standby threads is greater than the product;

incrementing the number of running threads using the running thread counter and decrementing the number of standby threads using the standby thread counter in response to a run request; and

updating the product set by the product running thread counter if the incremented number of running threads is greater than the product.

19. (CURRENTLY AMENDED) A controller for controlling a plurality of threads that perform parallel processing, comprising:

a thread management table storing thread information of the plurality of threads, wherein the thread information includes a number of running threads performing parallel processing and a number of standby threads that are in a standby state, wherein the thread management table includes

a running thread counter to count the number of running threads,

a standby thread counter to count the number of standby threads, and

a maximum running thread counter;

a thread management circuit requesting thread generation based on the number of standby threads stored in the thread management table, and requesting a standby thread to run;

a comparison circuit setting a maximum number of running threads during a predetermined period in accordance with the number of running threads included in the thread information to the maximum running thread counter, and comparing the number of standby threads with the maximum number of running threads;

a termination circuit terminating a number of standby threads exceeding the maximum number of running threads when the number of standby threads is greater than the maximum number of running threads; and

a thread to cause the running thread counter to increment the number of running threads and the standby thread counter to decrement the number of standby threads in response to a run request, wherein the thread updates the maximumnumber of running threads set by the maximum running thread counter if the incremented number of running threads is greater than the maximum number of running threads.

20. (CURRENTLY AMENDED) A controller for controlling a plurality of threads that perform parallel processing, comprising:

a thread management table storing thread information of the plurality of threads, wherein the thread information includes a number of running threads performing parallel processing and a number of standby threads that are in a standby state, wherein the thread management table includes

a running thread counter to count the number of running threads,

a standby thread counter to count the number of standby threads, and

an average running thread counter;

a thread management circuit requesting thread generation based on the number of standby threads stored in the thread management table, and requesting a standby thread to run;

a comparison circuit setting an average number of running threads during a predetermined time period in accordance with the number of running threads included in the thread information to the average running thread counter, and comparing the number of standby threads with the average number of running threads;

a termination circuit terminating a number of standby threads exceeding the average number of running threads when the number of standby threads is greater than the average number of running threads; and

a thread to cause the running thread counter to increment the number of running threads

and the standby thread counter to decrement the number of standby threads in response to a run request, wherein the thread updates the average number of running threads set by the average running thread counter if the incremented number of running threads is greater than the average number of running threads.

21. (PREVIOUSLY PRESENTED) A controller for controlling a plurality of threads that perform parallel processing, comprising:

a thread management table storing thread information of the plurality of threads,wherein the thread information includes a number of running threads performing parallel processing and a number of standby threads that are in a standby state, wherein the thread management table includes

a running thread counter to count the number of ruming threads,

a standby thread counter to count the number of standby threads, and

a product running thread counter;

a thread management circuit requesting thread generation based on the number of standby threads stored in the thread management table, and requesting a standby thread to run;

a comparison circuit setting a product to the product running thread counter, the product being obtained by multiplying the number of running threads during a predetermined time period by a predetermined coefficient in accordance with the number of running threads included in the thread information, and comparing the number of standby threads with the product;

a termination circuit terminating a number of standby threads exceeding the product when the number of standby threads is greater than the product; and

a thread to cause the running thread counter to increment the number of running threads and the standby thread counter to decrement the number of standby threads in response to a run request, wherein the thread updates the product set by the product running thread counter if the incremented number of running threads is greater than the product.

22. (CURRENTLY AMENDED) A computer readable storage medium storing a program for controlling at least one processor to execute a plurality of threads that perform parallel processing, according to a method comprising:

counting a number of running threads performing parallel processing using a running thread counter and a number of standby threads that are in a standby state using a standby thread counter for a predetermined time period;

setting a maximum number of running threads in accordance with the number of running

threads during the predetermined time period using a maximum running thread counter;

comparing the number of standby threads with the maximum number of running threads; ~~and~~

terminating an amount of the standby threads exceeding the maximum number of running threads when the number of standby threads is greater than the maximum number of running threads;

incrementing the number of running threads using the running thread counter and decrementing the number of standby threads using the standby thread counter in response to a run request; and

updating the maximum number of running threads set by the maximum running thread counter if the incremented number of running threads is greater than the maximum number of running threads.

23. (CURRENTLY AMENDED) A computer readable storage medium storing a program for controlling at least one processor to execute a plurality ofthreads that perform parallel processing, according to a method comprising:

counting a number of running threads performing parallel processing using a running thread counter and a number of standby threads that are in a standby state using a standby thread counter for a predetermined time period;

setting an average number of running threads in accordance with the number of running threads during the predetermined time period using an average running thread counter;

comparing the number of standby threads with the average number of running threads; ~~and~~

terminating an amount of the standby threads exceeding the average number of running threads when the number of standby threads is greater than the average number of running threads;

incrementing the number of running threads using the running thread counter and decrementing the number of standby threads using the standby thread counter in response to a run request; and

updating the average number of running threads set by the average running thread counter if the incremented number of running threads is greater than the average number of running threads.

24. (PREVIOUSLY PRESENTED) A computer readable storage medium storing a program for controlling at least one processor to execute a plurality of threads that perform parallel processing, according to a method comprising:

counting a number of running threads performing parallel processing using a running thread counter and a number of standby threads that are in a standby state using a standby thread counter for a predetermined time period;

setting a product obtained by multiplying the number of running threads during the predetermined time period by a predetermined coeficient using a product running thread counter;

comparing the number of standby threads with the product;

terminating an amount of the standby threads exceeding the product when the number of standby threads is greater than the product;

incrementing the number of running threads using the running thread counter and decrementing the number of standby threads using the standby thread counter in response to a run request; and

updating the product set by the product running thread counter if the incremented number of running threads is greater than the product.

To, Jennifer N.

From: AOlenginski@s-n-h.com
Sent: Tuesday, May 09, 2006 1:45 PM
To: To, Jennifer N.
Subject: Application Serial No. 09/748,862 - Attorney Docket No. 1076.1060

Examiner's
ıendment.c

Examiner To -

Attached are the proposed claim amendments per our telephone discussion of May 8, 2006.

Best regards,

Allison Olenginski
<<Examiner's Amendment.doc>>
mailto:aolenginski@s-n-h.com

Allison Olenginski
Staas & Halsey LLP
1201 New York Ave., N.W.
Suite 700
Washington, D.C. 20005
U.S.A.
Main Tel: 202.434.1500
Direct Tel: 202.454.1547
Fax: 202.434.1501

Our website is located at http://www.staasandhalsey.com